IN THE SPECIFICATION

Please disregard the title page of the application before page 1 of the application. Except for Steven Scott, the persons listed are not inventors in this application. Steven Scott executed a Declaration as a sole inventor.

On page 1, please amend this section as follows:

Related Applications

This application is related to ~~U.S. Patent Application No. ______________, entitled "Multistream Processing System and Method", filed on even date herewith; to U.S. Patent Application No. ______________, entitled "System and Method for Synchronizing Memory Transfers", Serial No. ______________, filed on even date herewith; to~~ U.S. Patent Application No.[[______________]] 10/643,742, entitled "Decoupled Store Address and Data in a Multiprocessor System", filed on even date herewith; to U.S. Patent Application No. [[______________]] 10/643586, entitled "Decoupled Vector Architecture", filed on even date herewith; to U.S. Patent Application No.[[______________]] 10/643,727, entitled ~~"Latency Tolerant Distributed Shared Memory Multiprocessor Computer"~~ "Method and Apparatus for Indirectly Addressed Vector Load-Add-Store Across Multi-Processors", filed on even date herewith; to U.S. Patent Application No.[[______________]] 10/643,754, entitled "Relaxed Memory Consistency Model", filed on even date herewith; to U.S. Patent Application No. [[______________]] 10/643,758, entitled "Remote Translation Mechanism for a Multinode System", filed on even date herewith; and to U.S. Patent Application No.[[______________]] 10/643,741, entitled "~~Method and Apparatus for Local Synchronizations in a Vector Processor System~~ Multistream Processing Memory-And Barrier-Synchronization Method And Apparatus", filed on even date herewith, each of which is incorporated herein by reference.

Please amend the paragraph beginning at line 30, page 2, as follows:

Figure 1shows one embodiment of an MSP 100. The MSP 100 includes two processors ~~110~~ 900 and two cache memories 120.

Please amend the paragraph beginning at line 11, page 3, as follows:

Figure 9 shows one embodiment of processor ~~110~~ 900. Each processor 110 is composed of a scalar processor 910 and two vector pipes 930. The scalar and vector unit are decoupled with respect to instruction execution and memory accesses. Decoupling with respect to instruction execution means the scalar unit can run ahead of the vector unit to resolve control flow issues and execute address arithmetic. Decoupling with respect to memory accesses means both scalar and vector loads are issued as soon as possible after instruction dispatch. Instructions that depend upon load values are dispatched to queues where they await the arrival of the load data. Store addresses are computed early and their addresses saved for later use. Each scalar processor 910 is capable of decoding and dispatching one vector instruction (and accompanying scalar operand) per cycle. Instructions are sent in order to the vector units, and any necessary scalar operands are sent later after the vector instructions have flowed through the scalar unit's integer or floating point pipeline and read the specified registers. Vector instructions are not sent speculatively; that is, the flow control and any previous trap conditions are resolved before sending the instructions to the vector unit. For a further description of decoupled vector architecture please refer to the U.S. patent application entitled "Decoupled Vector Architecture", filed on even date herewith, the description of which is hereby incorporated by reference.

Please amend the paragraph beginning at line 28, page 3, as follows:

In another embodiment, processor ~~110~~ 900 contains a cache memory 920 for scalar references only. Local MSP cache coherence is maintained by requiring all data in processor cache memory 920 to be contained in MSP cache memory 120.

Please amend the paragraph beginning at line 1, page 4, as follows:

Within MSP 100 in Figure 1, each cache memory 120 is shared by each processor ~~110~~ 900. Each cache memory includes two processor ports 140 to allow sharing by processors 900, and one memory port 130 for accessing local memory 1000. Thus each MSP 100 contains two local memory ports 130.

Please amend the paragraph beginning at line 5, page 4, as follows:

Figure 2 shows another embodiment of an MSP 200. The MSP 200 is composed of two processors 900 and two cache memories 120. Each cache memory 120 includes two processor ports 140 to allow sharing by each processor ~~110~~ 900 and also includes two memory ports 130 for addressing local memory 1000. Thus each MSP 200 contains four local memory ports 130.

Please amend the paragraph beginning at line 20, page 4, as follows:

Figure 5 shows one embodiment of a processing node 500. The processing node 500 includes two MSPs 100 each having two local memory ports 130, one I/O channel controller 510, and two local memories 1000. Each local memory includes two MSP ports 1010. Thus each processor ~~110~~ 900 in Figure 1 can access each local memory 1000 in Figure 5.

Please amend the paragraph beginning at line 24, page 4, as follows:

Figure 6 shows another embodiment of a processing node 600. The processing node 600 includes four MSPs 200 each having four local memory ports 130, one I/O channel controller 510, and four local memories 1000. Each local memory includes four MSP ports 1010. Thus each processor ~~110~~ 900 in Figure 2 has access to each local memory 1000 in Figure 6.

Please amend the paragraph beginning at line 29, page 4, as follows:

Figure 7 shows another embodiment of a processing node 700. The processing node 700 includes two MSPs 300 each having eight local memory ports 130, one I/O channel controller 510, and eight local memories 1000. Each local memory includes two MSP ports 1010. Thus each processor ~~110~~ 900 in Figure 3 has access to each local memory 1000 in Figure 7.

Please amend the paragraph beginning at line 4, page 5, as follows:

Figure 8 shows another embodiment of a processing node 800. The processing node 800 includes four MSPs 400 each having sixteen local memory ports 130, two I/O channel controllers 510, and sixteen local memories 1000. Each local memory includes four MSP ports 1010. Thus each processor ~~110~~ 900 in Figure 4 can access each local memory 1000 in Figure 8.